LA-UR- 03-2033

Approved for public release; distribution is unlimited.

Title: LEARNING WITH THE RATCHET ALGORITHM

Author(s): Donald R. Hush, 115295, CCS-3
James C. Scovel, 097403, CCS-3

Submitted to: International Conference on Computational Learning Theory

Learning with the Ratchet Algorithm

Don Hush and Clint Scovel
Modeling, Algorithms and Informatics Group, CCS-3
Mail Stop B265
Los Alamos National Laboratory
Los Alamos, NM 87545
(dhush,jcs)@lanl.gov

March 18, 2003

Abstract

This paper presents a randomized algorithm called `Ratchet` that asymptotically minimizes (with probability 1) functions that satisfy a *positive–linear–dependent* (PLD) property. We establish the PLD property and a corresponding realization of `Ratchet` for a generalized loss criterion for both linear machines and linear classifiers. We describe several learning criteria that can be obtained as special cases of this generalized loss criterion, e.g. classification error, classification loss and weighted classification error. We also establish the PLD property and a corresponding realization of `Ratchet` for the Neyman–Pearson criterion for linear classifiers. Finally we show how, for linear classifiers, the `Ratchet` algorithm can be derived as a modification of the `Pocket` algorithm.

1 Introduction

To motivate the concepts introduced in this paper we describe two learning problems which possess a common property that enables to unified approach to their algorithm development. We begin with some definitions. Let $z \in \mathbb{R}^m$ and $\omega \in \mathbb{R}^m$. We say that z is ω*–positive* if $\omega \cdot z > 0$ (where $\cdot$ is the usual inner product). Let $\mathcal{I}$ be a countably infinite set and consider a set $Z = \{(z_1, i_1), ..., (z_n, i_n)\} \subseteq \mathbb{R}^m \times \mathcal{I}$ where $z_j \in \mathbb{R}^m$ and $\{i_1, ..., i_n\} \subset \mathcal{I}$ (this definition allows the set Z to have repeated values of z distinguished by their index value i). We use the abbreviated notation $Z = \{z_{i_1}, ..., z_{i_n}\}$ for this set and we call this type of set a *multisample*. In addition we call $\{i_1, ..., i_n\}$ the index set for Z. Similarly we denote the subset $\{(z_j, i_j), ..., (z_k, i_k)\} \subseteq Z$ by $\{z_{i_j}, ..., z_{i_k}\}$ and refer to it as a *subsample* of Z with index set $\{i_j, ..., i_k\} \subseteq \{i_1, ..., i_n\}$. We define $Z^+ \subseteq Z$ to be a *positive linear* (PL) subsample of Z if there exists an $\omega \in \mathbb{R}^m$ such that all members of Z^+ are ω–positive, and define

$$\Omega^+ = \{\omega : \omega \cdot z_i > 0, \forall z_i \in Z^+\}$$

to be the witness set for Z^+. For technical reasons we define the empty set to be a PL multisample with the whole space as its witness set. Finally we define $P(Z) = \{Z_1^+, Z_2^+, ...\}$ to be the (finite) set of all PL subsamples Z_i^+ of Z.

In the first problem we are given a multisample $Z = \{z_1, ..., z_n\}$ and asked to determine a value $\omega \in \mathbb{R}^m$ that maximizes the criterion $R(\omega) = |\{z_i : \omega \cdot z_i > 0, i = 1, 2, ..., n\}|$. Geometrically we seek a hyperplane through the origin that dichotomizes $\mathbb{R}^m$ so that the number of samples in the positive half–space is maximized. This problem is closely related to the problem of determining a linear classifier that maximizes the number of correctly classified training samples for a two–class classification problem. Now consider the collection of all linear dichotomies of Z and the corresponding collection of PL subsamples of Z formed from the positive samples of these dichotomies. These PL subsamples form a subset of $P(Z)$ that accounts for all criterion values, that is the criterion values for this problem are *witnessed by* this subset of $P(Z)$.

In our second problem we are given a multisample $Z = \{z_{1,1}, z_{1,2}, z_{2,1}, z_{2,2}, ..., z_{n,1}, z_{n,2}\}$, where the samples are paired according to their first index, and we are asked to determine a value $\omega \in \mathbb{R}^m$ that maximizes the criterion $R(\omega) = |\{(z_{i,1}, z_{i,2}) : \omega \cdot z_{i,1} > 0 \text{ and } \omega \cdot z_{i,2} > 0, i = 1, 2, ..., n\}|$. Geometrically we seek a hyperplane through the origin that dichotomizes $\mathbb{R}^m$ so that the number of sample pairs in the positive half–space is maximized. This problem is closely related to the problem of determining a linear machine that maximizes the number of correctly classified training samples for a three–class classification problem (e.g. see (Cannon, Fugate, Hush, & Scovel, 2003)). Now define the *positively paired subsample* for a value ω to be $\{z_{i,j} \in Z : \omega \cdot z_{i,1} > 0 \text{ and } \omega \cdot z_{i,2} > 0\}$. This is the PL subsample of Z containing all the individual samples from ω–positive sample pairs. Now consider the collection of positively paired subsamples defined by all $\omega \in \mathbb{R}^m$. This collection forms a subset of $P(Z)$ that accounts for all criterion values, that is the criterion values for this problem are *witnessed by* this subset of $P(Z)$.

Although the witness subset for the first problem may be different than the second, the two problems above are similar in that their criterion values are witnessed by some subset of $P(Z)$. This paper describes several important learning problems share this property. In principle all such problems can be solved in a finite number of steps by searching over a finite subset of $\mathbb{R}^m$ that witnesses the members of $P(Z)$, but this approach is not efficient since $|P(Z)|$ can be exponential in the dimension m. Furthermore, the specific problems of interest here can all be shown to be NP–Hard. On the other hand, the first problem above can be solved by the `Pocket` algorithm (Gallant, 1990). `Pocket` is a simple randomized algorithm that produces an optimal solution asymptotically (with probability 1), and has proven to be effective in empirical studies where it is terminated after a finite number of steps (Burgess, Zenzo, & Granieri, 1992; Gallant, 1990; Golea & Marchand, 1990; Windeatt & Tebbs, 1997). In this paper we show how to modify the `Pocket` algorithm to obtain a simple randomized algorithm called `Ratchet` that visits every member of $P(Z)$ (asymptotically with probability 1) and therefore produces an optimal solution to any problem whose criterion values are witnessed by a subset of $P(Z)$.

Formally the `Ratchet` algorithm minimizes functions that satisfy a property we call *positive-linear–dependent* (PLD). Section 2 defines the PLD property and presents the `Ratchet` algorithm. To realize `Ratchet` for a particular function we must construct a map ϕ that witnesses the PLD property. Section 3 establishes maps ϕ that witness the PLD property for a generalized loss criterion for both linear machines and linear classifiers. Several important learning criteria are obtained as special cases of this generalized loss criterion. This section also shows how, for linear classifiers, the `Ratchet` algorithm can be derived as a modification of the `Pocket` algorithm (Gallant, 1990). In Section 4 we show how `Ratchet` can be applied to the Neyman–Pearson learning problem by establishing a map ϕ that witnesses the PLD property for the Neyman–Pearson criterion.

2 PLD Criteria and the Ratchet Algorithm

The `Ratchet` algorithm was introduced by Cannon et al. (Cannon et al., 2003) to solve a learning problem related to the task of selecting a restoration method for digitized documents in such a way that the average OCR error of the documents is reduced. This section summarizes the relevant results from that paper.

We consider minimization problems with criteria R that satisfy the following definition.

Definition 2.1. Let $\mathcal{A}$ be a set and let R be a function from $\mathcal{A} \times \mathbb{R}^m$ to $\mathbb{R}$. Suppose that for every $A \in \mathcal{A}$, $R_A = R(A, \cdot)$ achieves its infimum on a nontrivial set $\Omega^*(A) \subseteq \mathbb{R}^m$. Then R is a *positive–linear–dependent* (PLD) function if there exists a map to multisamples $\phi : \mathcal{A} \to\to \mathbb{R}^m \times \mathcal{I}$, such that for every $A \in \mathcal{A}$ there exists a PL subset of the multisample $\phi(A) = \{z_{i_1}, z_{i_2} ...\}, z_{i_j} \in \mathbb{R}^m, \{i_1, i_2, ...\} \subset \mathcal{I}$ whose witness set Ω^+ satisfies $\Omega^+ \subseteq \Omega^*(A)$.

In our application to learning problems $\mathcal{A}$ is the set of all training sets, $\mathbb{R}^m$ is the classifier parameter space, and R_A is an empirical error function that we wish to minimize with our choice of parameter $\omega \in \mathbb{R}^m$.

The `Ratchet` algorithm in Algorithm 1 is a simple algorithm for optimizing a PLD criterion when a map ϕ is known. This algorithm simply runs the randomize perceptron algorithm on the multisample $Z = \phi(A)$, computes the criterion value R_A each time ω changes value and saves the one with the smallest criterion value. The following theorem from Cannon, et al. (Cannon et al., 2003) establishes the optimality of this algorithm. The central idea in the proof of this theorem is to show that with probability 1 the ω visited by the randomized perceptron algorithm witness every PL subset of Z.

Theorem 2.1. *Let R be a PLD criterion witnessed by a map ϕ. For every $A \in \mathcal{A}$ consider the sequence $\omega(k), k = 0, 1, ...$ produced by the* `Ratchet` *algorithm with inputs A, R, ϕ. Let $\omega^*(k), k = 0, 1, ...$ be a sequence that satisfies $\omega^*(k) \in \arg\min_{\omega(i):i=0,1,...,k} R_A(\omega(i))$. Then*

$$R_A(\omega^*(k)) \overset{wp1}{\to} \min_{\omega} R_A(\omega)$$

where wp1 denotes "with probability 1".

To realize `Ratchet` for a particular PLD criterion we must construct a map ϕ that witnesses the PLD property. To assist in the determination of such a map, and in verification of the PLD property, the following lemma is established in Cannon et al. (Cannon et al., 2003). This lemma gives sufficient conditions that can be checked once a map ϕ has been proposed.

Lemma 2.1. *Let $\mathcal{A}$ be a set and let R be a function from $\mathcal{A} \times \mathbb{R}^m$ to $\mathbb{R}$. Suppose that for every $A \in \mathcal{A}$, $R_A = R(A, \cdot)$ achieves its infimum on a nontrivial set $\Omega^*(A) \subseteq \mathbb{R}^m$. Let $\phi : \mathcal{A} \to\to \mathbb{R}^m \times \mathcal{I}$ be a map to multisamples. For $A \in \mathcal{A}$ let $Z = \phi(A) = \{z_{i_1}, ..., z_{i_n}\}, z_{i_j} \in \mathbb{R}^m, \{i_1, ..., i_n\} \subset \mathcal{I}$ and let $J^+(\omega) = \{i_j : \omega \cdot z_{i_j} > 0\}$ denote the index set of ω–positive samples from Z. If for every $A \in \mathcal{A}$ and every $\omega \in \mathbb{R}^m$ there exists an $\acute{\omega} \in \mathbb{R}^m$ such that*

2.1.1. $J^+(\acute{\omega}) \supseteq J^+(\omega)$

2.1.2. $R_A(\acute{\omega}) = R_A(\omega)$

2.1.3. $\left(\omega_0, \omega_1 \in \mathbb{R}^m \text{ and } J^+(\acute{\omega}_0) \supseteq J^+(\acute{\omega}_1)\right) \Rightarrow \left(R_A(\omega_0) \leq R_A(\omega_1)\right)$.

then R is PLD witnessed by ϕ.

Algorithm 1 `Ratchet`: In practice ω^* is returned after a finite number of steps.

`INPUTS`: An element $A \in \mathcal{A}$, a criterion function R, and a map ϕ

{Compute the multisample Z}
$Z = \{z_{i_1}, ..., z_{i_n}\} \leftarrow \phi(A)$

{Initialize parameters.}
Set $\omega(0)$ and ω^* to zero and set $R^* \leftarrow R_A(\omega^*)$.

{Perform the randomized perceptron algorithm and track the best solution.}
$k \leftarrow 0$
loop
  $i \leftarrow$ random sample index drawn uniformly from $\{i_1, i_2, ..., i_n\}$
  if $(\omega(k) \cdot z_i \leq 0)$ **then**
    $\omega(k+1) \leftarrow \omega(k) + z_i$
    if $(R_A(\omega(k+1)) < R^*)$ **then**
      $R^* \leftarrow R_A(\omega(k+1))$
      $\omega^* \leftarrow \omega(k+1)$
    end if
  else
    $\omega(k+1) \leftarrow \omega(k)$
  end if
  $k \leftarrow k+1$
end loop

3 The Generalized Loss Criterion

In this section we determine computable maps ϕ that witness the PLD property for a generalized loss criterion. Through appropriate choices of a loss function we show how this criterion realizes several important criteria encountered in standard learning problems for linear machines and linear classifiers. In addition we show how, for linear classifiers, the `Ratchet` algorithm can be derived as a modification of the `Pocket` algorithm (Gallant, 1990).

3.1 The M–Class Problem with Linear Machines

Consider the following M–class learning problem. Let $\mathcal{N}$ be the set of natural numbers and define $\mathcal{A} = \cup_{k>0}(\mathbb{R}^d \times \mathbb{R}^M \times \mathcal{N})^k$ to be the set of training sets where the multisample $A = \{(x_1, l_1), ..., (x_n, l_n)\} \in \mathcal{A}$ is a training set with n samples, $x_i \in \mathbb{R}^d$ is the feature vector for the i–th sample and $l_i = (l_i(0), ..., l_i(M-1)) \in \mathbb{R}^M$ is the corresponding loss vector. The value $l_i(j)$ represents the loss incurred when x_i is assigned to class j. Let $\mathcal{M} = \{0, 1, ..., M-1\}$ and $\omega = (w_1, w_2, ..., w_M) \in \mathbb{R}^{M(d+1)}$ and consider the family of linear machines $f_\omega : \mathbb{R}^d \to \mathcal{M}$ defined by

$$f_\omega(x) = \max_{k \in \mathcal{K}_\omega(x)} k \tag{1}$$

where $\mathcal{K}_\omega(x)$ is the subset of $\mathcal{M}$ given by

$$\mathcal{K}_\omega(x) = \arg\max_{k\in\mathcal{M}} w_k \cdot (1, x). \tag{2}$$

The generalized loss criterion $R : \mathcal{A} \times \mathbb{R}^{d+1} \to \mathbb{R}$ is defined by

$$R_A(\omega) = \sum_{i=1}^{n} l_i(f_\omega(x_i)). \tag{3}$$

Important special cases of this criterion are obtained when $y_i \in \mathcal{M}$ is the correct class label for x_i and we set loss values as follows.

1. The *classification error* criterion is obtained by setting

   $$l_i(j) = I(j \neq y_i).$$

   where $I(\cdot)$ is the indicator function that takes a value 1 when its argument is true and 0 otherwise.

2. The *classification loss criterion* is obtained by setting

   $$l_i(j) = c(j, y_i)$$

   where c is a $M \times M$ loss matrix. This criterion is often employeed with the diagonal elemnts of c set to 0 (so that the loss for correct classification is 0). The off–diagonal elements represent losses for each of the $M(M-1)$ different error types. Setting $c(j,j) = 0, \forall j$ and $c(j,k) = 1, \forall j \neq k$ gives the *classification error* criterion above which is also called the "0-1" loss criterion.

The following theorem is proved in Cannon, et al. (Cannon et al., 2003).

Theorem 3.1. *The function $R : \mathcal{A} \times \mathbb{R}^{d+1} \to \mathbb{R}$ defined by (3) is PLD witnessed by the map ϕ in Definition 3.1 below.*

The map ϕ described here is an extension of Kesler's construction for the multiclass problem (see p. 266 in (Duda, Hart, & Stork, 2000), pp. 87–93 in (Nilsson, 1990), and (Smith, 1969)).

Definition 3.1. Let $\mathcal{Z} = (1 \times \mathbb{R}^d)^M$ and let $\rho : \mathbb{R}^d \to\to \mathcal{Z} \times \mathcal{N}^2$ be the map $\rho = \rho_2\rho_1$ where $\rho_1 : \mathbb{R}^d \to 1 \times \mathbb{R}^d$ is defined by $x \mapsto (1, x)$ and $\rho_2 : 1 \times \mathbb{R}^d \to\to \mathcal{Z} \times \mathcal{N}^2$ is the map to multisamples defined by

$$\xi \mapsto \{..., \zeta_{jk}, ...\}, \quad 1 \leq j \leq M, \ k : 1 \leq k \leq M, k \neq j$$

where $\zeta_{jk} \in \mathcal{Z}$ is the vector obtained by concatenating M vectors as follows: $\xi = (1, x)$ is placed in the j–th position, $-\xi$ in the k–th position, and zero vectors are placed in the other $M-2$ positions as illustrated below,

$$\zeta_{jk} = (0...0 \underbrace{\xi}_{j^{th}} 0...0 \underbrace{-\xi}_{k^{th}} 0...0).$$

Now define ζ_{ijk} to be the jk–th member of $\rho(x_i)$. Let $\epsilon > 0$ and define

$$\Delta_{ijk} = \begin{cases} \epsilon, & l_i(j) = l_i(k) \\ l_i(k) - l_i(j), & \text{otherwise} \end{cases}, \ 1 \leq i \leq n, 1 \leq j \leq M, k : 1 \leq k \leq M, k \neq j.$$

With $z_{ijk} = \Delta_{ijk}\zeta_{ijk}$ the map $\phi : \mathcal{A} \to\to \mathcal{Z} \times \mathcal{N}^3$ to multisamples is given by

$$\phi(A) = \{..., z_{ijk}, ...\}, \ \forall ijk \text{ such that } \Delta_{ijk} > 0.$$

3.2 The 2–Class Problem with Linear Classifiers

When $M = 2$ it is simpler to use a linear classifier than a 2–class linear machine. In this section we prove the PLD property for the general loss criterion over linear classifiers. As a consequence we obtain a map ϕ that is much simpler than Definition 3.1. In addition we show how, for this criterion, the `Ratchet` algorithm can be derived as a modification of the `Pocket` algorithm (Gallant, 1990).

We consider the same learning problem described in the previous section except that we restrict to $M = 2$ and we replace the class of linear machines with the class of linear classifiers $f_\omega : \mathbb{R}^d \to \{0, 1\}$ defined by

$$f_\omega(x) = \begin{cases} 0, & \omega \cdot (1, x) \leq 0 \\ 1, & \omega \cdot (1, x) > 0 \end{cases} \tag{4}$$

where $\omega \in \mathbb{R}^{d+1}$.

In addition to the special cases of the generalized loss criterion described in the previous section, a third case arises here. The *weighted classification error* criterion is obtained by setting

$$l_i(j) = \gamma_i I(j \neq y_i)$$

where $\gamma_i \geq 0, i = 1, 2, ..., n$ and $y_i \in \{0, 1\}$ is the correct class label for x_i. This criterion is encountered in many boosting algorithms. For example each round of the AdaBoost algorithm determines new values for $\gamma_i, i = 1, 2, ..., n$, and then seeks a base classifier that minimizes the corresponding weighted classification error (Freund & Shapire, 1997).

We now describe how the `Ratchet` algorithm can be derived as a modification of the `Pocket` algorithm. Gallant introduced `Pocket` to minimize the *classification error* criterion for linear classifiers. `Pocket` operates by running the randomized perceptron algorithm on the multisample $Z = \{z_1, ..., z_n\}, z_i = (2y_i - 1)(1, x_i)$, computing the *run length* for each ω visited (i.e. the number of consecutive ω–positive samples encountered before ω is modified by the algorithm), and retaining the $\omega(k)$ with the largest run length in the "pocket". Gallant also introduces a variation called `Pocket-with-Ratchet` that places a new value of ω in the pocket only when it has both a larger run length and witnesses a smaller criterion value. These `Pocket` algorithms are attractive because the run length is very simple to compute, but they may not be appropriate for the generalized loss criterion. For example consider the obvious adaptation of the `Pocket-with-Ratchet` algorithm that operates on the same multisample Z and replaces the value of ω in the pocket when the run length is larger and the criterion value $R_A(\omega)$ is smaller. With $l_i(j) = I(j \neq y_i)$ this criterion is minimized when the number of positive samples in Z is maximized and so values of ω with larger run lengths are more likely to have smaller criterion values, but this is not necessarily true for the generalized loss. In fact it seems unlikely that any statistic computed on ω–positive samples only can be used to order the classifier space for the generalized loss. More generally the determination of a suitable replacement for the run length rule remains an open problem. The `Ratchet` algorithm is obtained by removing the run length rule from `Pocket-with-Ratchet` so that a value of ω with the smallest criterion value is saved in the pocket. This requires that the criterion value be computed each time ω is modified and therefore requires more computation than the `Pocket` algorithms, but it yields a viable algorithm. Indeed, Theorem 3.2 below verifies that the generalized loss criterion for linear classifiers is PLD witnessed by a map that gives $z_i = (l_i(0) - l_i(1))(1, x_i)$.

Theorem 3.2. *The function $R : \mathcal{A} \times \mathbb{R}^{d+1} \to \mathbb{R}$ defined by (3) with $M = 2$ and f_ω defined by (4) is PLD witnessed by the map $\phi : \mathcal{A} \to\!\!\to \mathbb{R}^{d+1} \times \mathcal{N}$ defined by $\phi(\{(x_1, l_1), ..., (x_n, l_n)\}) = \{z_1, ..., z_n\}, z_i = (l_i(0) - l_i(1))(1, x_i)$.*

Proof. For any $A \in \mathcal{A}$ and any $\omega \in \mathbb{R}^{d+1}$ the criterion value is a finite sum and therefore the criterion achieves its infimum on a nontrivial set $\Omega^*(A) \subseteq \mathbb{R}^{d+1}$. Define $\xi_i = (1, x_i)$ and write

$$\begin{aligned} R_A(\omega) &= \sum_{i=1}^{n} l_i(f_\omega(x_i)) \\ &= \sum_{i=1}^{n} l_i(0) I(\omega \cdot \xi_i \leq 0) + l_i(1) I(\omega \cdot \xi_i > 0). \end{aligned}$$

Define $\Delta_i = l_i(0) - l_i(1)$ and write the criterion value as

$$\begin{aligned} R_A(\omega) \ &= \sum_{i=1}^{n} \Big(I(\Delta_i > 0) \ \ (l_i(0) - |\Delta_i| I(\omega \cdot \xi_i > 0)) \ + \\ &\qquad I(\Delta_i < 0) \ \ (l_i(1) - |\Delta_i| I(\omega \cdot \xi_i \leq 0)) \ + \ I(\Delta_i = 0) \ l_i(0) \Big) \\ &= \sum_{i=1}^{n} \max(l_i(0), l_i(1)) - |\Delta_i| \Big(I(\Delta_i > 0, \omega \cdot \xi_i > 0) + I(\Delta_i < 0, \omega \cdot \xi_i \leq 0) \Big) \\ &= \sum_{i=1}^{n} \max(l_i(0), l_i(1)) - |\Delta_i| \Big(I(\Delta_i \neq 0, \Delta_i \omega \cdot \xi_i > 0) + I\big(\Delta_i < 0, \Delta_i \omega \cdot \xi_i = 0\big) \Big). \end{aligned}$$

The definition of ϕ gives $z_i = \Delta_i \xi_i$ so that

$$\begin{aligned} R_A(\omega) \ &= \sum_{i=1}^{n} \max(l_i(0), l_i(1)) - |\Delta_i| \Big(I(\Delta_i \neq 0, \omega \cdot z_i > 0) + I\big(\Delta_i < 0, \omega \cdot z_i = 0\big) \Big) \\ &= C - |\Delta_i| \Big(I(\Delta_i \neq 0, \omega \cdot z_i > 0) + I\big(\Delta_i < 0, \omega \cdot z_i = 0\big) \Big). \end{aligned} \tag{5}$$

where $C = \sum_{i=1}^{n} \max(l_i(0), l_i(1))$. To complete the proof we verify conditions 2.1.1-2.1.3 in Lemma 2.1. Let $Z = \{z_1, ..., z_n\}$. For any $\omega \in \mathbb{R}^{d+1}$ let

$$\delta = \begin{cases} 1, & \omega \cdot z_i = 0 \text{ for all } z_i \in Z \\ \min_{z_i \in Z, \omega \cdot z_i \neq 0} |\omega \cdot z_i|, & \text{otherwise} \end{cases}$$

and let

$$\acute{\omega} = \omega - (\delta/2, 0), \quad 0 \in \mathbb{R}^d.$$

This gives

$$\begin{array}{ll} \acute{\omega} \cdot z_i \geq |\Delta_i|\delta/2 > 0, & \text{when } (\Delta_i \neq 0, \omega \cdot z_i > 0) \text{ or } (\omega \cdot z_i = 0, \Delta_i < 0) \\ \acute{\omega} \cdot z_i \leq -|\Delta_i|\delta/2 < 0, & \text{when } (\Delta_i \neq 0, \omega \cdot z_i < 0) \text{ or } (\omega \cdot z_i = 0, \Delta_i > 0) \end{array}$$

and therefore condition 2.1.1 holds and (5) can be written

$$R_A(\omega) = R_A(\acute{\omega}) = C - \sum_{i=1}^{n} |\Delta_i| I(\Delta_i \neq 0, \acute{\omega} \cdot z_i > 0) = C - \sum_{i \in J^+(\acute{\omega})} |\Delta_i|.$$

which verifies condition 2.1.2. The right hand side of this expression also establishes a monotonic relation between nested sets J^+ and the values of R_A. This verifies condition 2.1.3 and completes our proof. ♦

4 The Neyman–Pearson Criterion

The Neyman–Pearson problem is a 2–class problem where the goal is to maximize the correct classification for one class subject to an upper bound on the classification error for the other class. Cannon et al. (Cannon, Howse, Hush, & Scovel, 2002b) describe a learning strategy for the Neyman–Pearson problem that determines a classifier from sample data by solving a constrained optimization problem. We restrict to linear classifiers and reformulate this constrained optimization problem as an unconstrained optimization problem. We then provide a simple map ϕ that witnesses the PLD property for the unconstrained optimization criterion.

Define $\mathcal{A} = \cup_{k>0}(\mathbb{R}^d \times \{0,1\} \times \mathcal{N})^k$ to be the set of training sets where the multisample $A = \{(x_1, y_1), ..., (x_n, y_n)\} \in \mathcal{A}$ is a training set with n samples, $x_i \in \mathbb{R}^d$ is the feature vector for the i–th sample and $y_i \in \{0,1\}$ is the corresponding class label. Let $f_\omega : \mathbb{R}^d \to \{0,1\}$ be the class of linear classifiers defined by (4). The fraction of samples from class 0 that are correctly classified by f_ω is denoted

$$c_0(f_\omega) = \frac{1}{n_0} \sum_{i:y_i=0} I(f_\omega(x_i) = 0).$$

and the fraction of samples from class 1 that are incorrectly classified by f_ω is denoted

$$e_1(f_\omega) = \frac{1}{n_1} \sum_{i:y_i=1} I(f_\omega(x_i) \neq j),$$

where n_j is the number of samples with $y = j$. If $n_0 = 0$ then we define $c_0(f_\omega) = 1$ and if $n_1 = 0$ we define $e_1(f_\omega) = 0$. The Neyman–Pearson learning strategy chooses a classifier that solves the constrained optimization problem (Cannon et al., 2002b)

$$\begin{array}{ll} \max_{\omega \in \mathbb{R}^{d+1}} & c_0(f_\omega) \\ \text{subject to} & e_1(f_\omega) \leq \alpha \end{array} \tag{6}$$

where $\alpha \geq 0$. A solution to this problem always exists because the set of linear classifiers that satisfy the constraint is nontrivial for any training set $A \in \mathcal{A}$. Thus we can reformulate (6) as the following unconstrained optimization problem

$$\min_{\omega \in \mathbb{R}^{d+1}} -c_0(f_\omega) + p(e_1(f_\omega) - \alpha) \tag{7}$$

where the penalty function p is defined by

$$p(\theta) = \begin{cases} 0, & \theta \leq 0 \\ \infty, & \theta > 0. \end{cases}$$

Consequently the Neyman–Pearson criterion $R : \mathcal{A} \times \mathbb{R}^{d+1} \to \mathbb{R}$ is defined by

$$R_A(\omega) = -c_0(f_\omega) + p(e_1(f_\omega) - \alpha). \tag{8}$$

The following theorem provides a simple map ϕ that witnesses the PLD property for this criterion.

Theorem 4.1. *The function $R : \mathcal{A} \times \mathbb{R}^{d+1} \to \mathbb{R}$ defined by (8) is PLD witnessed by the map $\phi : \mathcal{A} \to\to \mathbb{R}^{d+1} \times \mathcal{N}$ defined by $\phi(\{(x_1, y_1), ..., (x_n, y_n)\}) = \{z_1, ..., z_n\}, z_i = (2y_i - 1)(1, x_i)$.*

Proof. This proof is structured similarly to the proof of Theorem 3.2. For any $A \in \mathcal{A}$ and any $\omega \in \mathbb{R}^{d+1}$ the criterion value is a finite sum and therefore the criterion achieves its infimum on a nontrivial set $\Omega^*(A) \subseteq \mathbb{R}^{d+1}$. Define $\xi_i = (1, x_i)$ and write

$$\begin{aligned} R_A(\omega) &= -c_0(f_\omega) + p(e_1(f_\omega) - \alpha) \\ &= -\frac{1}{n_0} \sum_{i:y_i=0} I(\omega \cdot \xi_i \leq 0) + p\left(\frac{1}{n_1} \sum_{i:y_i=1} I(\omega \cdot \xi_i \leq 0) - \alpha\right) \end{aligned}$$

Now rewrite the argument of the penalty function in terms correctly classified samples.

$$R_A(\omega) = -\frac{1}{n_0} \sum_{i:y_i=0} I(\omega \cdot \xi_i \leq 0) + p\left(1 - \alpha - \frac{1}{n_1} \sum_{i:y_i=1} I(\omega \cdot \xi_i > 0)\right).$$

The definition of ϕ gives $z_i = -\xi_i$ when $y_i = 0$, and $z_i = \xi_i$ when $y_i = 1$, which yields

$$R_A(\omega) = -\frac{1}{n_0} \sum_{i:y_i=0} I(\omega \cdot z_i \geq 0) + p\left(1 - \alpha - \frac{1}{n_1} \sum_{i:y_i=1} I(\omega \cdot z_i > 0)\right) \tag{9}$$

To complete the proof we verify conditions 2.1.1-2.1.3 in Lemma 2.1. Let $Z = \{z_1, ..., z_n\}$. For any $\omega \in \mathbb{R}^{d+1}$ let

$$\delta = \begin{cases} 1, & \omega \cdot z_i = 0 \text{ for all } z_i \in Z \\ \min_{z_i \in Z, \omega \cdot z_i \neq 0} |\omega \cdot z_i|, & \text{otherwise} \end{cases}$$

and let

$$\acute{\omega} = \omega - (\delta/2, 0), \quad 0 \in \mathbb{R}^d.$$

This gives

$$\begin{array}{ll} \acute{\omega} \cdot z_i \geq \delta/2 > 0, & \text{when } (\omega \cdot z_i > 0) \text{ or } (\omega \cdot z_i = 0, y_i = 0) \\ \acute{\omega} \cdot z_i \leq -\delta/2 < 0, & \text{when } (\omega \cdot z_i < 0) \text{ or } (\omega \cdot z_i = 0, y_i = 1) \end{array}$$

and therefore condition 2.1.1 holds and (5) can be written

$$R_A(\omega) = R_A(\acute{\omega}) = -\frac{1}{n_0} \sum_{i:y_i=0} I(\acute{\omega} \cdot z_i > 0) + p\left(1 - \alpha - \frac{1}{n_1} \sum_{i:y_i=1} I(\acute{\omega} \cdot z_i > 0)\right)$$

which verifies condition 2.1.2. Defining $J_0^+(\acute{\omega}) = \{i \in J^+(\acute{\omega}) : y_i = 0\}$ and $J_1^+(\acute{\omega}) = \{i \in J^+(\acute{\omega}) : y_i = 1\}$ gives

$$R_A(\omega) = -\frac{1}{n_0}|J_0^+(\acute{\omega})| + p\left(1 - \alpha - \frac{1}{n_1}|J_1^+(\acute{\omega})|\right)$$

The first term on the right side is monotonically decreasing in $|J_0^+(\acute{\omega})|$ and the penalty term is monotonically decreasing in $|J_1^+(\acute{\omega})|$. Since $J^+(\acute{\omega}) = J_0^+(\acute{\omega}) \cup J_1^+(\acute{\omega})$, the right side of this expression establishes a monotonic relation between nested sets J^+ and the values of R_A. This verifies condition 2.1.3 and completes our proof. ♦

References

Burgess, N., Zenzo, S. D., & Granieri, M. N. (1992). The generalization of a constructive algorithm in pattern classification problems. *International Journal of Neural Systems, 3*, 1–6.

Cannon, A., Howse, J., Hush, D., & Scovel, C. (2002b). *Learning with the Neyman-Pearson and min-max criteria* (Los Alamos Technical Report Nos. LA-UR-02-2951). Los Alamos National Laboratory. (submitted for publication, http://wwwc3.lanl.gov/ml/pubs_select.shtml)

Cannon, M., Fugate, M., Hush, D., & Scovel, C. (2003). Selecting a restoration technique to minimize ocr error. *accepted for publication in IEEE Transactions on Neural Networks.* (see also Los Alamos Technical Report: LAUR-01-6860 at http://wwwc3.lanl.gov/ml/pubs_ml.shtml.)

Duda, R., Hart, P., & Stork, D. (2000). *Pattern classification.* New York, NY: Wiley.

Freund, Y., & Shapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139.

Gallant, S. (1990). Perceptron–based learning algorithms. *IEEE Transactions on Neural Networks, 1*(2), 179–191.

Golea, M., & Marchand, M. (1990). A growth algorithm for neural network decision trees. *EuroPhys. Lett., 12*, 205.

Nilsson, N. (1990). *The mathematical foundation of learning machines.* San Mateo, CA: Morgan–Kaufmann.

Smith, F. (1969). Design of multicategory pattern classifiers with two–category classifier design procedures. *IEEE Transactions on Computers, C–18*, 548–551.

Windeatt, T., & Tebbs, R. (1997). Spectral technique for hidden layer neural network training. *Pattern Recognition Letters, 18*, 723–731.